International Journal of Science and Research Archive

eISSN: 2582-8185

Cross Ref DOI: 10.30574/ijsra

Journal homepage: https://ijsra.net/

(REVIEW ARTICLE)

Check for updates

An Optimal Artificial Intelligence (AI) technique for cybersecurity threat detection in IoT Networks

Mani Gopalsamy *

Senior Cyber Security Specialist, Louisville, KY, USA- 40220.

International Journal of Science and Research Archive, 2022, 07(02), 661–671

Publication history: Received on 03 October 2022; revised on 16 December 2022; accepted on 20 December 2022

Article DOI: https://doi.org/10.30574/ijsra.2022.7.2.0235

Abstract

An exponential growth rate has been seen in cyberattacks targeting fully integrated servers, apps, and communications networks. The Things Network (IoT). Inefficient operation of sensitive devices harms end users, increasing the risk of identity theft and cyberattacks, increasing costs, and decreasing revenue as problems with the Internet of Things network remain undetected for long periods. Robust cybersecurity solutions are necessary to safeguard digital infrastructures against the growing frequency of cyberattacks and the fast growth of the Internet of Things. This research looks at the function of Artificial Intelligence (AI) in improving cybersecurity measures, specifically emphasising the comparison of signature-based and anomaly-based IDS. ML and DL techniques, including DNN, SVM, and Random Forest classifiers, are used in this work to classify cybersecurity risks and detect potential threats using the dataset UNSW-NB15. According to our data, the Random Forest model outperforms the competition, with a 98.6% accuracy rate and 99% precision, F1 score and recall. The research emphasises the efficacy of AI-powered systems in real-time threat identification, emphasising its usefulness in advancing cybersecurity measures. By tackling the issues provided by conventional security measures and employing modern ML and DL approaches, this study gives significant insights for organisations trying to improve their cybersecurity policies in an increasingly complex threat scenario.

Keywords: Cybersecurity; Artificial Intelligence; Machine Learning; Threat Detection Systems; Internet of Things; UNSW-NB15.

1. Introduction

Throughout the past several years, the prevalence of IoT devices in daily life has increased significantly. System safety measures are receiving more attention as the role of information technology in people's daily lives grows in response to growing concerns about network security and privacy[1]. There have been increased attempts to compromise computer systems and machine networks due to the proliferation of sophisticated technology and new Internet applications, such as the IoT[2][3]. The IoT has seen rapid growth. It consists of smart gadgets and networked things that run without direct human involvement. Many smart Internet of Things devices have sensors built into them that make it easier for them to connect to the Internet. This allows information to be shared across different nodes for applications in transportation, healthcare, agriculture, and other fields [4][5]. IoT devices aim to transform work habits[6], save time and resources, and optimise operations[7]. Not only can the IoT be very advantageous, but it also offers a boundless amount of opportunities for the distribution, expansion, and personalisation of data[8].

Most cybersecurity specialists will always consider IoT devices as fortunately, never to be targeted again because, in this one weakness, attackers can easily overpower them[9]. It has also contributed to the escalation of artificial intelligence being used by hackers to bypass other computational systems developed to assist in detecting such strange behaviour. The emergence of IoT technology has resulted in lots of attention being paid to AI[10]. As a result of this

* Corresponding author: Mani Gopalsamy

Copyright © 2022 Author(s) retain the copyright of this article. This article is published under the terms of the Creative Commons Attribution Liscense 4.0.

expansion, IoT cybersecurity apps have started to integrate AI instruments such as DT, linear regression, and ML in the quest for identifying cyber threats [11].

One of society's major turning points is the appearance of AI as a driving force in development. Compared to the more traditional security approaches to security, then AI offers much more credible results. This is a great example of AI power; it can analyse and learn from voluminous databases and new information and predict perils[12][13]. The cyber threat environment is dynamic and hence cannot be overcome by traditional measures such as through firewalls, antivirus and other programs. There is an obvious demand for such characteristics as innovative, strong, and efficient cybersecurity solutions[14][15]. It is also important with high-speed capabilities that immediate alerts on any suspicious activities or anomalies that are detected are provided by the machine learning models. AI can recognise emergent, malicious behaviour on the network, for example, an increase in data processing or login activity, and present them as security threats[16]. This means that real-time analysis enables organisational security personnel to respond to potential security incidents faster, hence minimising their effects and slashing the time required to address perceived security threats by more than half[17].

Specifically, the purpose of the present study is to review and comparatively analyse a number of approaches and techniques of both ML and DL for enhancing cybersecurity in general, with a focus on such subfields as cybersecurity threat analysis and intrusion detection. This study tries to identify and categorise cybersecurity threats more accurately using advanced computation methods; it seeks to meet the growing challenge of threats in a connected world where IoT devices have gained popularity.

1.1. Contributions of the Study

This work enriches cybersecurity knowledge stock significantly since it explores the DL and ML-based methodologies for handling and analysing network attacks employing the UNSW-NB15 set. The key contributions are as:

- Through feature scale, missing value and one hot encoder, the study maintains the quality data that results in enhanced performance of the developed models.
- Evaluate various models, including CNN, ANN, LSTM, and Random Forest, providing a comparative analysis for detecting cybersecurity threats.
- The use of the dataset UNSW-NB15 ensures the study's relevance by incorporating real-world data that reflects current cyber threats.
- Utilising metrics like recall, accuracy, precision, and F1-score to assess model performance to offer comprehensive insights into the efficacy of different approaches

1.2. Organization of the paper

The paper's structure is as follows: Previous studies on cybersecurity threats, together with any gaps in the literature and new additions, are presented in Sections I and II. The mechanism of this is then provided in Section III. Section V offers a conclusion and future scope, whereas Section IV presents findings and discusses them. Conclusion and future endeavours are presented in the final section.

2. Literature Review

The previous research on Cybersecurity Threat Detection in IoT Networks employing ML and DL techniques is provided in this section.

In this study, Kodali and Muntean (2021) compare how to distinguish between attack and regular network data using cutting-edge DL models like Autoencoder-FCN and FCN. The CICIDS2017 dataset, which comprises further than 2.8 million network data annals and reflects real-world data, is used in the study; the dataset reflects the furthermost current, common outbreaks observed in current network settings and includes both typical and attack data. Because of low error rates and accuracy parameters greater than 97%, FCN and Autoencoder-FCN were observed to work very well. When comparing the two approaches, the FCN model finds it simpler to get somewhat better performance than the model of Autoencoder-FCN. The autoencoder-FCN model took longer to train when the model was deployed remotely, but the FCN model took less time[18].

This paper by, Mosaiyebzadeh et al., (2021) proposes a NIDS that is developed using DL and which takes is input from a dataset of MQTT attacks freely available to the public. Recall, Accuracy, precision, F1-score, and weighted average are a few of the metrics that are typically with the gauge how effective the proposal is. Through study of our DL-based

Network IDS's performance assessment findings, we were able to identify MQTT assaults with an F1 sof 98.33% and an average accuracy of 97.09%. The current research is extremely replicable because the experiments are also available on GitHub. This makes sense as well because the trials were also shared on GitHub Asia [19].

In this study, Xu et al., (2021) introduce a new method for detecting network anomalies that employ five layers of autoencoder (AE). The rationale for our proposing this is based on the results that we obtained after carrying out a detailed and very comprehensive analysis of several performance parameters linked to an AE model for predicting breast cancer. In order to minimise the effect of imbalance between feature set data types in the model's bias, in this research work, we provide a novel approach to data pre-processing that reorganises and eliminates input samples that include more significant outliers. Thus, for the identification of typical or abnormal network traffic samples, our suggested model employs the best reconstruction error function. Applying those advanced techniques and the proper model structure, our model is crucially capable of learning features and performing dimensional reduction successfully, which contributes to raising the accuracy of detection and, therefore, the f1-score. In this paper, we applied the NSL-KDD data set to examine the feasibility of the model we proposed; the current methods were surpassed by those in terms of highest detection accuracy and f1 of 90.61% and 92.26%, individually [20].

In this work, Farhin et al. (2020) propose a paradigm for identifying attacks on the IoT using software-defined networks. The SDN controller reported that traffic and source nodes could be prohibited after the flow of traffic has been studied, or abnormality is spotted. With SDN, a system which implements FNNs to detect some forms of assaults such as; man-in-the-middle, DDoS, side-channel, and harmful code is being contemplated. Two datasets from NSL-KDD are employed to train and evaluate the FNN. The consequences of the tests establish that the optional attack detection system, which uses FNN, has an accuracy rate of 83% in identifying the specified assault[21].

In this paper, Alrashdi et al., (2019) one of the solutions put forward is the AD-IoT system, a smart anomaly detector in smart cities and the RF algorithm is used for ML. It is noted that distributed fog nodes of the system employing the proposed approach can detect hacked IoT devices. To evaluate and demonstrate the correctness of the model, a modern dataset was employed. The findings suggest that the AD-IoT may complete the lowest FPR and the best classification accuracy of 99.34%[22].

Table 1 Summary of Comparative Analysis of Cybersecurity Threat Detection in IoT Networks

References	Methodology	Dataset	Performance	Limitations & Future Work
Mosaiyebzadeh et al. [19]	Network IDS based on Deep Learning	Public dataset using MQTT	97.09% Accuracy, 98.33% F1-Score	DL-based Network IDS shows high accuracy and F1-score in detecting MQTT attacks, with shared code for reproducibility.
Kodali and Muntean[18]	FCN and Autoencoder-FCN models for detection of intrusion are compared.	CICIDS2017 (more than 2.8M records of network data)	FCN and Autoencoder-FCN both achieved over 97% accuracy; FCN had lower training time.	Further optimisation needed for Autoencoder-FCN to reduce training time.
Alrashdi et.al. [22]	The RF method is used by the Anomaly detection IoT (AD-IoT) system to identify anomalies at fog nodes.	Modern IoT cybersecurity dataset	Classification accuracy: 99.34%, Lowest false positive rate.	More verification is required on bigger and more varied datasets.
Farhin et al. (2020) [21]	Fuzzy Neural Network (FNN) in Software-defined Networks (SDN)	NSL-KDD	83% Accuracy	FNN-based attack detection system in SDN effectively detects multiple attack types with moderate accuracy.
Xu et al. (2021) [20]	5-layer autoencoder model with new preprocessing methods and reconstruction error function.	NSL-KDD	Accuracy: 90.61%, F1-Score: 92.26%	Limited to NSL-KDD; may not generalise to other datasets.

2.1. Research gaps

Several research gaps exist for IoT contexts, despite advances in IDS and machine learning-based anomaly detection. Current research has displayed promising outcomes according to accuracy and performance, such as the high accuracy of signature-based IDS, anomaly-based systems, and different ML models, including FCN, RF, and GB approaches. However, the trade-offs between energy efficiency, computing resource use, and real-world application have yet to be adequately resolved. For example, anomaly-based IDS use much more power, and models like Autoencoder-FCN have long training durations, which might be restricting in resource-constrained IoT applications.

3. Methodology

This section outlines the suggested approach for controlling and analysing cybersecurity threats, taking into account all available ML and DL techniques. The analysis of the UNSW-NB15 dataset starts with data collection, followed by extensive preparation, which includes feature normalisation to standardise data, addressing missing values, and using One-Hot Encoding for category labelling. The data flow in various steps and phases that shown in data flow Figure 1. Feature selection is done to save just the most important characteristics, which improves model performance and computational efficiency. Then, two sets are created from the dataset: 20% is used for testing and 80% for training. On the basis of the data, many classification models are trained, including RF, LSTM, ANN, and CNN. These models are evaluated for their capacity to identify network intrusions using a variation of crucial performance criteria, such as precision, recall, accuracy and F1-score, which provide dependable and accurate predictions.

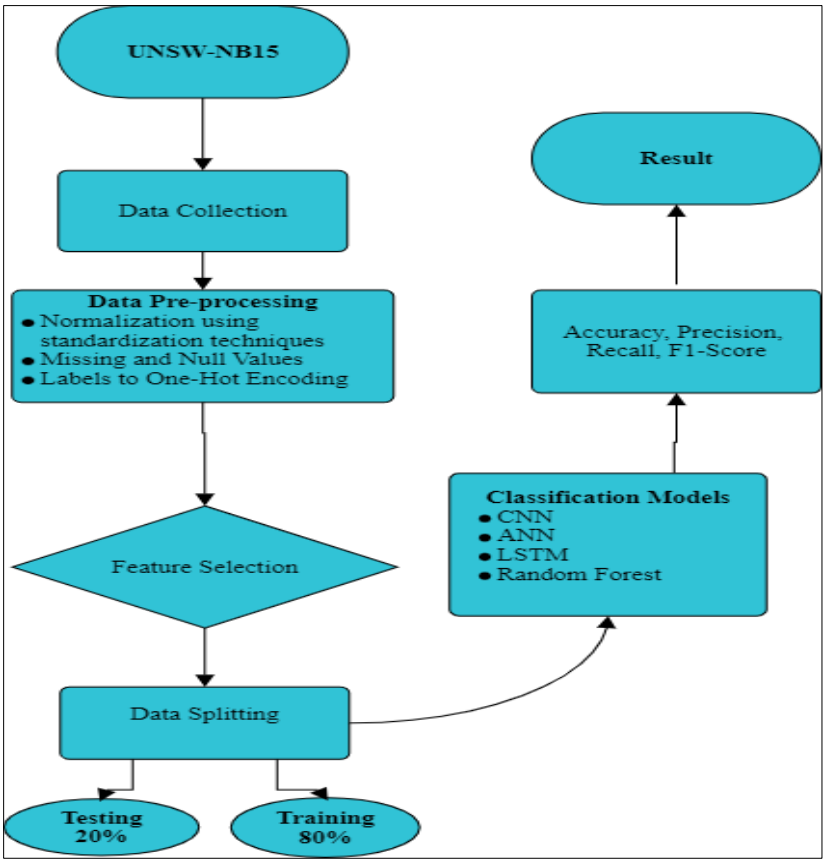

Figure 1 Data Flow Diagram

The following steps of a data flow diagram are briefly explained below:

3.1. Data Collection

The UNSW-NB15 dataset was used to carry out this investigation. The test set has 82,332 records, whereas the training set contains 175,341 values. The collection includes both typical network traffic and nine distinct types of attacks. From the network packets, 42 characteristics have been retrieved. Numerous approaches are used to categorise attacks, including worms, reconnaissance, Shellcode, fuzzers, analysis, backdoors, denial of service, exploits, and generics.

3.2. Data Preprocessing

The implementation process includes data preparation as one of its primary components. An essential initial step in the information discovery process is data pre-processing. There are several procedures involved, such as transformation and reduction of data. Improving the quality of the raw data is necessary to guarantee the efficacy and precision of learning algorithms. Thus, by following the relevant data preparation procedures and utilising the appropriate learning algorithms, the collected data may be appropriately examined. [23]. Further processing key terms are as follows

- **Missing and Null Values:** the columns or rows that contain zeros or null values in order to remove the missing values.
- **Drop columns:** delete and drop the unnecessary columns from the dataset.

3.3. Normalization using standardisation

The features were changed by using the standardisation method. Normalisation is a data transformation procedure that centres on a zero mean and a one standard deviation. Limits are not bound by this procedure. In Equation (1), we can see a standardisation formula in which x stands for an observation, representing both the data's mean and standard deviation.

$$x_n = \frac{x-\mu}{\sigma} \qquad (1)$$

where xn = normalized value, x = original value, μ = mean of data, and σ = data standard deviation.

3.4. One-Hot Encoding for data labeling

A kind of hot encoding is one that uses binary representation of the data as a feature. This is a popular approach that compares the numerical variable at each level to a predetermined baseline. This thesis represents the acquired data set using binary vectors representing category variables using one hot encoding [24].

3.5. Feature Selection

An essential part of data preparation is feature selection. Improving model performance and reducing complexity may be achieved by identifying and choosing key characteristics. In this phase, the Pearson correlation technique was utilised to identify highly correlated features in the dataset, helping to select the most relevant ones for the analysis [25].

3.6. Data Splitting

Before starting to build the model, separate the data into subsets for testing and training. Eighty per cent of the dataset was utilised for training, while 20 per cent was used for testing the model.

3.7. Classification Models

This section discusses the Analysis and Classification of ML models like SVM, DNN, and RF explained below:

3.7.1. Support vector machine (SVM)

Currently, the most widely utilised ML technology by data scientists and enterprises worldwide is SVM, a robust learning algorithm. Hyperplanes are used to divide the various classes in this supervised learning technique, which searches for patterns within classes. In order to efficiently divide the features into discrete classes, the technique entails projecting the given characteristics into a feature space that is high-dimensional and optimising the hyperplane.

3.7.2. Deep Neural Networks (DNN)

An ANN with input, hidden, and output layers is called a DNN. By adjusting the relative importance of its connections, this network architecture may learn to do distributed processing in parallel [26].

3.7.3. Random Forest (RF)

An ML technique called RF belongs to the class of parallel ensembling techniques, which we shall discuss in a moment. It expands upon the idea of parallel tree training known as bagging. Bagging uses data produced via bootstrap aggregating, which takes many random samples and replaces them with data from the original collection, to construct

trees. All of the trees ˆf are trained using each subsample B. Thus, every tree takes in information from a number of subsamples that are somewhat different from one another.

$$\hat{f}_{bag}(x) = \frac{1}{B}\sum_{b=1}^{B}\hat{f}^{*b}(x) \quad (2)$$

The last forecast in a regression tree is just the mean of all the projections, as shown in (2) [27]. In order to reduce variation and improve generalisation error, bagging may be used to create numerous decision trees on the same set of data. As a result of using the same variables in every regression tree, bagging results in a high correlation between the trees. Consequently, this will exacerbate the generalisation mistake and raise the variance. This issue is resolved in random forests by decorating the trees. If just m factors are considered at each split, the trees will all have different properties and be constructed from different samples. Following the random generation of many features (j) at each split, the feature with the splitting point t that minimises (1) is chosen. The least significant factors have an opportunity to affect the outcome since every tree is different and has randomised properties, which implies that the most significant variable is not necessarily at the first split. The model will thus function better on data that is not in the sample. A few hyperparameters are also used by the random forest method to reduce overfitting to training data. These establish the maximum depth of each tree, the number of trees to be considered at each split, and the number of trees utilised in the model.

3.8. Model Evaluation

A performance matrix that contrasted the actual observations with the model projections was used to assess the effectiveness of the chosen models. The metrics in the performance matrix were recall, F1-score, precision, and accuracy. Each class's metrics were calculated independently: True Positives (TPs) are the number of positive events that were correctly recognised, while True Negatives (TNs) are the number of negative events that were correctly categorised. There are two types of incorrect classifications: false positives (FPs) and false negatives (FNs). FPs show how many occurrences were wrongly classed as positive, while FNs show how many instances were wrongly classified as negative. Using these formulae, we can express the assessment metrics:

3.8.1. Accuracy

The most popular and straightforward indicator for evaluating models is accuracy, which gives a clear indication of the percentage of samples that are properly identified. The formula (3)

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (3)$$

3.8.2. Precision

The percentage of all true positives that fall inside the predicted positive range relative to all true positives is known as a precision measure. To illustrate, consider Equation 4: it represents the likelihood of accurately classifying a positive sample.

$$Precision = \frac{TP}{TP+FP} \quad (4)$$

3.8.3. Recall

The percentage of truly positive samples compared to the overall number of positive samples is called recall, which is often referred to as the detection rate. Equation 5 illustrates how the recall detection rate serves as a gauge for the model's capacity to identify attacks.

$$Recall = \frac{TP}{TP+FN} \quad (5)$$

3.8.4. F1-Score

The weighted average of recall (R) and precision (P) determines the machine learning F1-Score, with 0 being the lowest score and 1 the highest. Equation 6 illustrates the F1 Measure, a more thorough evaluation statistic than accuracy[28]:

$$F1 - Score = \frac{2*Precision*Recall}{(Precision+Recall)} \quad (6)$$

4. Result Analysis and Discussion

The result Analysis and Classification of Cybersecurity Threat Based on ML and DL. This analysis conducted, four evaluation metrics that were performed— Recordings of the confusion matrix, F1-Score, accuracy, recall, and precision were made, and they were used to assess deep learning models.

4.1. Exploratory Data Analysis (EDA)

A thorough understanding of the data requires the completion of exploratory analysis. A variety of procedures have been established to determine the optimal parameter values and extract relevant information. Initially, as shown in Figure 2, a heatmap was made to ascertain the level of association between various parameters.

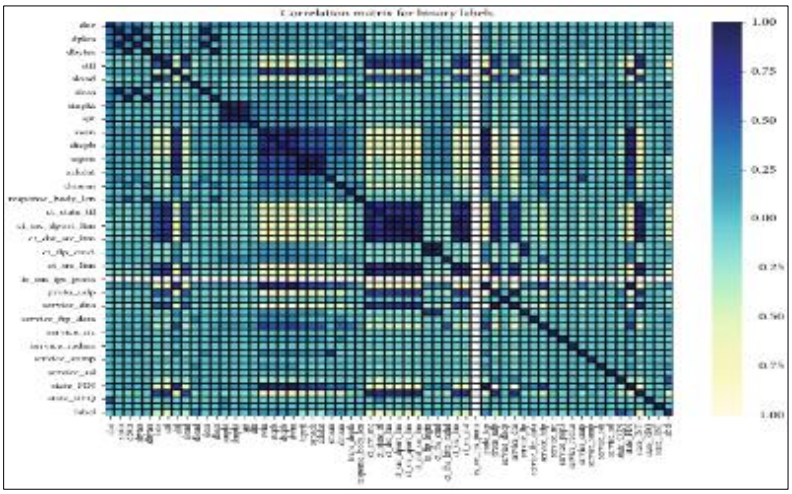

Figure 2 Heatmap for binary classification.

The correlation matrix for binary classification is seen in Figure 2. The heatmap's cells, which range in colour from dark blue (strong positive correlation) to dark red (strong negative correlation), each reflect the correlation coefficient between two variables. The correlation values vary from -1.00 to 1.00, according to the scale on the right.

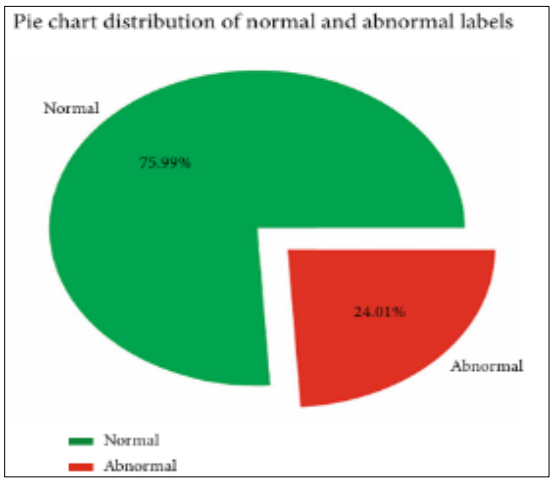

Figure 3 Dataset distribution

The normal and abnormal label distribution in a dataset is shown in Figure 3. For 75.19% of the data, the category is "Normal" (green portion), while for 24.01% of the data, it is "Abnormal" (red section). This shows that the majority of the occurrences in the dataset are usual, with just a small fraction being classified as aberrant.

4.2. Experiment results

This section provides the experiment result of ML model for Cybersecurity Threat Detection in IoT Networks. Table 2 shows the RF model achieves highest accuracy.

Table 2 Results of the Random Forest model on UNSW-NB15 dataset

Matrix	Random forest
Accuracy	0.986
Precision	0.99
Recall	0.99
F1-Score	0.99

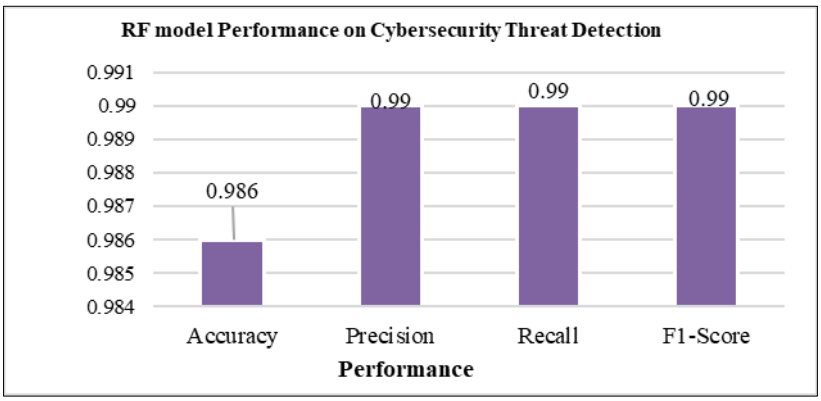

Figure 4 Random Forest Performance on Dataset

The bar graph for the RF model is shown in Table 2 and Figure 4. The RF model achieves 0.986 accuracy, recall, precision, and f1-score is 0.99%.

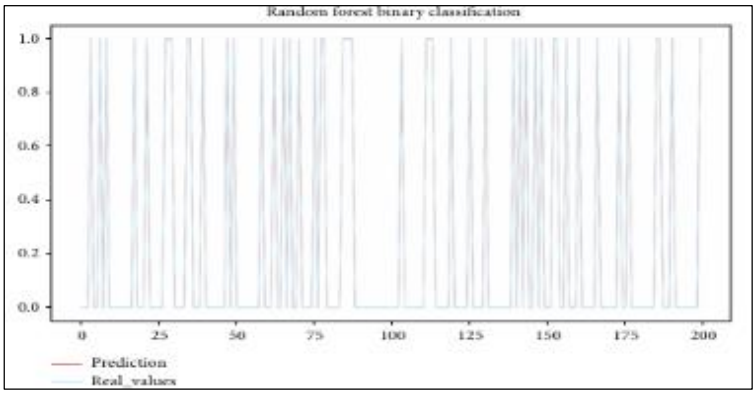

Figure 5 Random Forest Classifier results

A graph Random Forest binary classification is seen in Figure 5. It does a series of comparisons between actual values (in blue) and predicted values (in red). For binary outcomes, the x-axis may take values between 0 and 200, while the y-axis can take values between 0.0 and 1.0. To evaluate how well an RF algorithm categorises binary outcomes, this graphical depiction compares predicted values to actual ones.

4.3. Comparative analysis

The comparison between RF and another machine learning model performance on UNSW-NB-15 data across performance parameters. This comparison shows the RF model outperforms compare to other models, as shown in Table 3.

Table 3 Comparative analysis for cybersecurity threat detection on dataset

Models	Accuracy	Precision	Recall	F1-Score
SVM[29]	62.42	60.91	60.91	60.91
DNN[30]	75.9	81.9	72.4	76.6
RFC	98.6	99	99	99

The comparative analysis of Cybersecurity Threat detection with ML and DL models is displayed in Table 3. The comparison of SVM, DNN, and RF Classifier models shows a clear distinction in their performance. SVM exhibits the lowest performance crosswise all systems of metrics, with an accuracy of 62.42% and equally low recall, precision, and F1-score of 60.91%, indicating moderate prediction capabilities. In contrast, DNN significantly improves on these results, achieving an accuracy of 75.9%, with a notable increase in precision of 81.9%, though recall of 72.4% and F1-score 76.6% are slightly lower, reflecting a better balance between false positives and false negatives. However, RFC outperforms both models, demonstrating exceptional accuracy of 98.6% and achieving near-perfect precision, recall, and F1-scores of 99%, making it the most effective model in this comparison for accurate classification.

5. Conclusion

An exponential growth rate has been seen in the number of cyberattacks targeting fully integrated servers, apps, and communications networks via the IoT. End users are hurt by inefficient sensitive device operation, which in turn raises cyber dangers and identity abuse, drives up expenses, and cuts into income as issues with the IoT network go unnoticed for extended periods of time. Almost real-time monitoring of IoT interface attacks is required for effective safety and security. A Cybersecurity Threat Detection system that is designed to identify assaults that target the IoT has been created. This study uses UNSW-NB15 dataset for cyber threat detection. Important performance metrics, including precision, accuracy, F1-score and recall, were used to train and assess the models. The models were ready for use by carefully preparing the data, which included selecting and normalising the features. Because of its 98.6% accuracy, 99% precision, and 99% recall, the RF model outperformed all of the others when it came to NID. This study underscores the probable of RF and DL models in enhancing cybersecurity through accuracy and reliability. To enhance the robustness and adaptability of the models, future work could explore additional datasets with more diverse and dynamic attack patterns to improve generalisation across different network environments.

Compliance with ethical standards

Disclosure of conflict of interest

No conflict of interest is to be disclosed.

References

[1] A. N. Jahromi, H. Karimipour, A. Dehghantanha, and K. K. R. Choo, "Toward Detection and Attribution of Cyber-Attacks in IoT-Enabled Cyber-Physical Systems," *IEEE Internet Things J.*, 2021, doi: 10.1109/JIOT.2021.3067667.

[2] M. M. Rashid, J. Kamruzzaman, M. M. Hassan, T. Imam, and S. Gordon, "Cyberattacks detection in iot-based smart city applications using machine learning techniques," *Int. J. Environ. Res. Public Health*, 2020, doi: 10.3390/ijerph17249347.

[3] S. Pandey, "LEVERAGING WORKDAY FOR EFFECTIVE COVID-19 VACCINATION TRACKING: INTEGRATING CUSTOM OBJECTS AND SECURITY FEATURES IN HUMAN CAPITAL MANAGEMENT SYSTEMS," *Int. J. Bus. Quant. Econ. Appl. Manag. reseacrh*, vol. 7, no. 1, pp. 56–63, 2021.

[4] K. V. V. N. L. Sai Kiran, R. N. K. Devisetty, N. P. Kalyan, K. Mukundini, and R. Karthi, "Building a Intrusion Detection System for IoT Environment using Machine Learning Techniques," in *Procedia Computer Science*, 2020. doi: 10.1016/j.procs.2020.04.257.

[5] R. Goyal, "THE ROLE OF REQUIREMENT GATHERING IN AGILE SOFTWARE DEVELOPMENT: STRATEGIES FOR SUCCESS AND CHALLENGES," *Int. J. Core Eng. Manag.*, vol. 6, no. 12, pp. 142–152, 2021.

[6] J. Thomas, K. V. Vedi, and S. Gupta, "The Effect and Challenges of the Internet of Things (IoT) on the Management of Supply Chains," *Int. J. Res. Anal. Rev.*, vol. 8, no. 3, pp. 874–879, 2021.

[7] P. Pathak, A. Shrivastava, and S. Gupta, "A survey on various security issues in delay tolerant networks," *J Adv Shell Program.*, vol. 2, no. 2, pp. 12–18, 2015.

[8] V. V. Kumar, A. Sahoo, and F. W. Liou, "Cyber-enabled product lifecycle management: A multi-agent framework," in *Procedia Manufacturing*, 2019. doi: 10.1016/j.promfg.2020.01.247.

[9] V. K. Y. Nicholas Richardson, Rajani Pydipalli, Sai Sirisha Maddula, Sunil Kumar Reddy Anumandla, "Role-Based Access Control in SAS Programming: Enhancing Security and Authorization," *Int. J. Reciprocal Symmetry Theor. Phys.*, vol. 6, no. 1, pp. 31–42, 2019.

[10] T. G. Zewdie and A. Girma, "IOT SECURITY AND THE ROLE OF AI/ML TO COMBAT EMERGING CYBER THREATS IN CLOUD COMPUTING ENVIRONMENT," *Issues Inf. Syst.*, 2020, doi: 10.48009/4_iis_2020_253-263.

[11] M. Kuzlu, C. Fair, and O. Guler, "Role of Artificial Intelligence in the Internet of Things (IoT) cybersecurity," *Discov. Internet Things*, vol. 1, no. 1, pp. 1–14, 2021, doi: 10.1007/s43926-020-00001-4.

[12] J. Thomas, K. V. Vedi, and S. Gupta, "Enhancing Supply Chain Resilience Through Cloud-Based SCM and Advanced Machine Learning: A Case Study of Logistics," *J. Emerg. Technol. Innov. Res.*, vol. 8, no. 9, 2021.

[13] A. P. A. Singh, "Streamlining Purchase Requisitions and Orders : A Guide to Effective Goods Receipt Management," *J. Emerg. Technol. Innov. Res.*, vol. 8, no. 5, pp. g179–g184, 2021.

[14] K. Aswal, A. Rajmohan, T. R. C. Akhil, S. Mukund, J. Vinitha Panicker, and J. P. Dhivvya, "Kavach: A Machine Learning based approach for enhancing the attack detection capability of firewalls," in *2021 12th International Conference on Computing Communication and Networking Technologies, ICCCNT 2021*, 2021. doi: 10.1109/ICCCNT51525.2021.9579836.

[15] K. Patel, "Quality Assurance In The Age Of Data Analytics: Innovations And Challenges," *Int. J. Creat. Res. Thoughts*, vol. 9, no. 12, pp. f573–f578, 2021.

[16] M. R. Kishore Mullangi, Vamsi Krishna Yarlagadda, Niravkumar Dhameliya, "Integrating AI and Reciprocal Symmetry in Financial Management: A Pathway to Enhanced Decision-Making," *Int. J. Reciprocal Symmetry Theor. Phys.*, vol. 5, no. 1, pp. 42–52, 2018.

[17] M. M. Watney, "Artificial intelligence and its' legal risk to cybersecurity," in *European Conference on Information Warfare and Security, ECCWS*, 2020. doi: 10.34190/EWS.20.026.

[18] S. K. Kodali and C. H. Muntean, "An Investigation into Deep Learning Based Network Intrusion Detection System for IoT Systems," in *Proceedings of 2021 IEEE International Conference on Data Science and Computer Application, ICDSCA 2021*, 2021. doi: 10.1109/ICDSCA53499.2021.9650111.

[19] F. Mosaiyebzadeh, L. G. Araujo Rodriguez, D. Macedo Batista, and R. Hirata, "A Network Intrusion Detection System using Deep Learning against MQTT Attacks in IoT," in *Proceedings - 2021 IEEE Latin-American Conference on Communications, LATINCOM 2021*, 2021. doi: 10.1109/LATINCOM53176.2021.9647850.

[20] W. Xu, J. Jang-Jaccard, A. Singh, Y. Wei, and F. Sabrina, "Improving Performance of Autoencoder-Based Network Anomaly Detection on NSL-KDD Dataset," *IEEE Access*, 2021, doi: 10.1109/ACCESS.2021.3116612.

[21] F. Farhin, I. Sultana, N. Islam, M. S. Kaiser, M. S. Rahman, and M. Mahmud, "Attack Detection in Internet of Things using Software Defined Network and Fuzzy Neural Network," in *2020 Joint 9th International Conference on Informatics, Electronics and Vision and 2020 4th International Conference on Imaging, Vision and Pattern Recognition, ICIEV and icIVPR 2020*, 2020. doi: 10.1109/ICIEVicIVPR48672.2020.9306666.

[22] I. Alrashdi, A. Alqazzaz, E. Aloufi, R. Alharthi, M. Zohdy, and H. Ming, "AD-IoT: Anomaly detection of IoT cyberattacks in smart city using machine learning," in *2019 IEEE 9th Annual Computing and Communication Workshop and Conference, CCWC 2019*, 2019. doi: 10.1109/CCWC.2019.8666450.

[23] Q. Li *et al.*, "Using fine-tuned conditional probabilities for data transformation of nominal attributes," *Pattern Recognit. Lett.*, 2019, doi: 10.1016/j.patrec.2019.08.024.

[24] K. Potdar, T. S., and C. D., "A Comparative Study of Categorical Variable Encoding Techniques for Neural Network Classifiers," *Int. J. Comput. Appl.*, 2017, doi: 10.5120/ijca2017915495.

[25] J. Li *et al.*, "Feature selection: A data perspective," *ACM Computing Surveys*. 2017. doi: 10.1145/3136625.

[26] X. Zhou, A. K. Qin, M. Gong, and K. C. Tan, "A Survey on Evolutionary Construction of Deep Neural Networks," *IEEE Trans. Evol. Comput.*, 2021, doi: 10.1109/TEVC.2021.3079985.

[27] L. Breiman, "Bagging predictors," *Mach. Learn.*, 1996, doi: 10.1007/bf00058655.

[28] H. Sayadi *et al.*, "Towards accurate run-time hardware-assisted stealthy malware detection: A lightweight, yet effective time series cnn-based approach†," *Cryptography*, vol. 5, no. 4, pp. 1–25, 2021, doi: 10.3390/cryptography5040028.

[29] S. M. Kasongo and Y. Sun, "Performance Analysis of Intrusion Detection Systems Using a Feature Selection Method on the UNSW-NB15 Dataset," *J. Big Data*, 2020, doi: 10.1186/s40537-020-00379-6.

[30] J. V. R. and D. S. B. R. Kumar, "Intrusion Detection On The Unsw-Nb15 Dataset Using Feature Selection And Machine Learning Techniques," *Int. J. Comput. Sci. Netw. Secur.*, vol. 18, no. 4, pp. 691–699, 2021.